

1 K
AF

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:

Barbara A. Christensen et al.

Serial No.: 10/848,748

Examiner: J. Wong

Filing Date: May 19, 2004

Group Art Unit: 2168

For: METHOD AND APPARATUS FOR CONTROLLING RESULT DATASET GENERATION
IN A JAVASCRIPT ENVIRONMENT

Docket No.: 33012/383/101

TRANSMITTAL SHEET

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

CERTIFICATE UNDER 37 C.F.R. 1.8: I hereby certify that this correspondence and the documents described herein are being deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to the: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on this 15th day of December, 2008

By: Carolyn I. Erickson

We are transmitting herewith the attached:

- ☒ A check in the amount of \$540.00 is enclosed.
- ☐ Small entity status of this application under 37 C.F.R. 1.9 and 1.27 has been established.
- ☒ Other: Appeal Brief in Triplicate.
- ☒ Please charge any deficiencies or credit any over payment in the enclosed fees to Deposit Account 14-0620.

By: Wayne A. Sivertson

Wayne A. Sivertson
Reg. No. 25,645

NAWROCKI, ROONEY & SIVERTSON, P.A.
Suite 401, Broadway Place East
3433 Broadway Street N.E.
Minneapolis, Minnesota 55413
Telephone: (612) 331-1464
Facsimile: (612) 331-2239



P A T E N T

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of)
Barbara A. Christensen et al.) Examiner J. Wong
Serial No. 10/848,748) Group Art Unit 2168
Filing Date: 05/19/04) Docket No. 33012/383/101
For: METHOD AND APPARATUS)
FOR CONTROLLING RESULT) APPEAL BRIEF
DATASET GENERATION IN A)
JAVASCRIPT ENVIRONMENT)

APPELLANT'S APPEAL BRIEF
FILED UNDER 37 C.F.R. § 41.37

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

CERTIFICATE UNDER 37 C.F.R. 1.8: I hereby certify that this correspondence is being deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to the: Commissioner for Patents, Alexandria, VA, 22313-1450 on this

15 th day of December, 2008.

By Carolyn I. Erickson

This appeal brief is being filed in triplicate within sixty days of the Notice of Appeal filed October 15, 2008. Permission is

hereby granted to charge or credit deposit account number 14-0620 for any errors in fee calculation. Appellants request this Appeal Brief be made of record and fully considered.

REAL PARTY IN INTEREST

The Real Party in interest is:

Unisys Corporation

Township Line and Union Meeting Roads

Blue Bell, Pennsylvania 19424

being the assignee of the entire right, title, and interest by all inventors, by way of assignment documents filed at Reel 016327, frame 0319, in the United States Patent and Trademark Office.

RELATED APPEALS AND INTERFERENCES

There are no known pending Appeals and/or Interferences which will directly affect or be directly affected by or have a bearing on the Board's decision in this appeal. Therefore, there are no decisions to be placed in the attached Related Proceedings Appendix.

TABLE OF CONTENTS

TABLE OF CONTENTS	3
STATUS OF CLAIMS	7
STATUS OF THE AMENDMENTS	8
SUMMARY OF CLAIMED SUBJECT MATTER	9
GROUND OF REJECTION TO BE REVIEWED ON APPEAL	23
ARGUMENT	24
I. Claim 11 is not objectionable for "informal invocation of "means-plus-function" language.	24
II. Claims 1-21 are not unpatentable under 35 U.S.C. 102(b) as anticipated by Shappir.	25
II.A. Claim 1 is not anticipated by Shappir	27
II.B. Claim 2 is not anticipated by Shappir	29
II.C. Claim 3 is not anticipated by Shappir	30
II.D. Claim 4 is not anticipated by Shappir	30

II.E. Claim 5 is not anticipated by Shappir .	31
II.F. Claim 6 is not anticipated by Shappir .	31
II.G. Claim 7 is not anticipated by Shappir .	33
II.H. Claim 8 is not anticipated by Shappir .	33
II.I. Claim 9 is not anticipated by Shappir .	33
II.J. Claim 10 is not anticipated by Shappir .	33
II.K. Claim 11 is not anticipated by Shappir .	34
II.L. Claim 12 is not anticipated by Shappir .	36
II.M. Claim 13 is not anticipated by Shappir .	36
II.N. Claim 14 is not anticipated by Shappir .	36
II.O. Claim 15 is not anticipated by Shappir .	37
II.P. Claim 16 is not anticipated by Shappir .	37
II.Q. Claim 17 is not anticipated by Shappir .	38
II.R. Claim 18 is not anticipated by Shappir .	38
II.S. Claim 19 is not anticipated by Shappir .	39
II.T. Claim 20 is not anticipated by Shappir .	39
II.U. Claim 21 is not anticipated by Shappir .	39

III. Claims 1-20 are not unpatentable over Rangnekar in view of Connor.	40
III.A. Claim 1 is not unpatentable over Rangnekar in view of Connor	44
III.B. Claim 2 is not unpatentable over Rangnekar in view of Connor	47

III.C. Claim 3 is not unpatentable over Rangnekar in view of Connor	48
III.D. Claim 4 is not unpatentable over Rangnekar in view of Connor	49
III.E. Claim 5 is not unpatentable over Rangnekar in view of Connor	49
III.F. Claim 6 is not unpatentable over Rangnekar in view of Connor	50
III.G. Claim 7 is not unpatentable over Rangnekar in view of Connor	51
III.H. Claim 8 is not unpatentable over Rangnekar in view of Connor	52
III.I. Claim 9 is not unpatentable over Rangnekar in view of Connor	52
III.J. Claim 10 is not unpatentable over Rangnekar in view of Connor	52
III.K. Claim 11 is not unpatentable over Rangnekar in view of Connor	53
III.L. Claim 12 is not unpatentable over Rangnekar in view of Connor	53
III.M. Claim 13 is not unpatentable over Rangnekar in view of Connor	54
III.N. Claim 14 is not unpatentable over Rangnekar in view of Connor	54

III.O. Claim 15 is not unpatentable over Rangnekar in view of Connor	55
III.P. Claim 16 is not unpatentable over Rangnekar in view of Connor	55
III.Q. Claim 17 is not unpatentable over Rangnekar in view of Connor	56
III.R. Claim 18 is not unpatentable over Rangnekar in view of Connor	57
III.S. Claim 19 is not unpatentable over Rangnekar in view of Connor	57
III.T. Claim 20 is not unpatentable over Rangnekar in view of Connor	57
IV. Claim 21 is not unpatentable over Rangnekar in view of Connor and further in view of Braddy.	58
CONCLUSION	60
CLAIMS APPENDIX	61
EVIDENCE APPENDIX	68
RELATED PROCEEDINGS APPENDIX	69

STATUS OF CLAIMS

The subject patent application was filed on May 19, 2004 containing claims 1-21. In accordance with the amendments filed September 25, 2006, January 12, 2007 May 8, 2007, June 8, 2007, November 9, 2007, and March 27, 2008, claims 1-21, being all pending claims, were amended. On July 16, 2008, the Examiner mailed a final office action again rejecting all pending claims. No amendment after final was filed in response thereto. Instead, Applicants filed a Notice of Appeal on October 15, 2008 appealing from the final rejection of claims 1-21, being all pending claims. Therefore, claims 1-21, being all pending claims, stand finally rejected and are presented in the Claims Appendix, hereto attached, in the form pending following entry of the amendment filed March 27, 2008. No pending claim has ever been found to contain allowable subject matter.

There remains a provisional obviousness-type double patenting rejection involving some but not all of the pending claims. This rejection is provisional, because no claim so rejected has been found allowable. Therefore, the rejection is not yet ripe. Applicants have previously offered and continue to offer to deal

with this issue by way of terminal disclaimer or other appropriate measure after the matter becomes ripe.

STATUS OF THE AMENDMENTS

Amendments were filed on September 25, 2006, January 12, 2007, May 8, 2007, June 8, 2007, November 9, 2007, and March 27, 2008. No amendment after final under 37 C.F.R. 1.116 was filed in response to the final official action mailed July 16, 2008. Thus, all requested amendments to the claims have been entered.

SUMMARY OF CLAIMED SUBJECT MATTER¹

The present invention generally relates to legacy data base management systems and more particularly relates to enhancements for providing access to such legacy data base management systems using a standardized object-based programming language to efficiently provide resultant reports². The present invention overcomes the disadvantages of the prior art by providing a method of and apparatus for efficiently utilizing the power of a full featured legacy data base management system by a user at a terminal coupled to the world wide web or Internet using a standardized object-based command language. In order to permit any such access, the present invention must first provide a user interface, called a gateway, which translates transaction data transferred from the user over the Internet in HTML format into a format from which data base management system commands and inputs may be generated. The gateway must also convert the data base management system responses and outputs into an HTML document for display on the user's Internet terminal. Thus, as a minimum, the gateway must make these format and protocol conversions. In the preferred embodiment, the

¹ The references to the specification and drawings provided herein are only exemplary and are not deemed to be limiting. The purpose of the references is to enable the Board to more quickly determine where the claimed subject matter is described within the present application.

²See Specification at page 1, lines 17-20.

gateway resides in the web server coupled to the user via the world wide web and coupled to proprietary data base management system.³

To make access to a proprietary legacy data base by Internet users practical, a sophisticated security system is required to prevent intentional or inadvertent unauthorized access to the sensitive data of an organization. As discussed above, such a security system should provide multiple levels of access to accommodate a variety of authorized user categories. In the preferred embodiment of the present invention, rather than defining several levels of data classification, the different classes of users are managed by identifying a security profile as a portion of those service requests requiring access to secure data. Thus, the security profile accompanies the data/service to be accessed. The user simply need provide a user-id which correlates to the access permitted. This permits certain levels of data to be accessed by one or more of the several classes of user.⁴

In the preferred mode of practicing the present invention, each user-id is correlated with a security profile. Upon preparation of the service request which provides Internet access to a given portion of the data base, the service request developer specifies which security profiles are permitted access to the data or a portion thereof. The service request developer can

³See Specification at page 7, lines 2-12.

⁴See Specification at page 7, lines 13-22.

subsequently modify the accessibility of any security profile. The utility of the system is greatly enhanced by permitting the service request developer to provide access to predefined portions of the data, rather than being limited to permit or deny access to all of the data involved.⁵

Whereas the gateway and the security system are the minimum necessary to permit the most rudimentary form of communication between the Internet terminal of the user and the proprietary data base management system, as explained above, the Internet is a "stateless" communication system; the addition of the gateway and the security system do not change this statelessness. To unleash the real power of the data base management system, the communication protocol between the data base and the user requires functional interaction between the various data transfers.⁶

The present invention adds state management to this environment. Instead of considering each transfer from the Internet user coupled with the corresponding server response as an isolated transaction event as defined by the world wide web, one or more related service requests may be functionally associated in a service request sequence as defined by the data base management system into a dialog.⁷

⁵See Specification at page 8, lines 1-7.

⁶See Specification at page 8, lines 8-14.

⁷See Specification at page 8, lines 15-19.

A repository is established to store the state of the service request sequence. As such, the repository can store intermediate requests and responses, as well as other data associated with the service request sequence. Thus, the repository buffers commands, data, and intermediate products utilized in formatting subsequent data base management service requests and in formatting subsequent HTML pages to be displayed to the user.⁸

The transaction data in HTML format received by the server from the user, along with the state information stored in the repository, are processed by a service handler into a sequence of service requests in the command language of the data base management system. Sequencing and control of the data base management system is via an administration module.⁹

Through the use of the repository to store the state of the service request sequence, the service handler to generate data base management command language, and the administration module , the world wide web user is capable of performing each and every data base management function available to any user, including a user from a proprietary terminal having a dedicated communication link which is co-located with the proprietary data base management system hardware and software. In addition, the data base management system user at the world wide web terminal is able to

⁸See Specification at page 8, line 20, through page 9, line 2.

⁹See Specification at page 9, lines 3-6.

accomplish this in the HTML protocol, without extensive training concerning the command language of the data base management system.¹⁰

In accordance with the preferred embodiment of the present invention, a new command, @SPI (stored procedure interface) is defined for the Business Information Server (BIS)/Cool ICE system. The new command has two primary modes of operation. First, the command provides the ability to execute a specified stored procedure and return the results. This includes the handling of rowsets, input variables, output variables, and input/output variables. Secondly, the command provides a method to query and return meta-data about stored procedures in a data base catalog. The meta-data will provide the available stored procedures as well as information about the parameters for the stored procedures.¹¹

Meta-data is data about data. It is a way of documenting datasets. The information contained in meta-data documents the creation of a dataset and gives an idea of what the cartographic product to which it is attached was designed to do.¹²

Rowsets are the central objects that enable DB (data base) components to expose and manipulate data in tabular form. A rowset object is a set of rows in which each row has columns of data. For

¹⁰See Specification at page 9, lines 7-14.

¹¹See Specification at page 9, lines 15-22.

¹²See Specification at page 10, lines 1-3.

example, providers present data, as well as meta-data, to consumers in the form of rowsets. Query processors present query results in the form of rowsets. The use of rowsets throughout data base systems makes it possible to aggregate components that consume or produce data through the same object.¹³

Without the present invention, the user must write the C code and make the proper API (Application Program Interface) calls to execute the stored procedure as well as handle input, output, and input/output variables. This is a difficult process and requires in depth knowledge of the data base API interface, in addition to the pitfalls of having to develop application code (memory allocation, pointer manipulation, configuring enough variable space, handling input/output variables, etc.). In addition to writing the application code and submitting the proper stored procedure command, users previously had no real mechanism to manipulate any data that is retrieved from the data source.¹⁴

The present invention provides users the ability to execute a specified stored procedure as well as handle rowsets, input variables, output variables, and input/output variables without having to develop the application code themselves. Developing the code is a very cumbersome process with a lot of room for errors.

¹³See Specification at page 10, lines 4-9.

¹⁴See Specification at page 10, lines 10-17.

Furthermore, the developer must be very knowledgeable concerning the API interface in order to correctly make proper calls.¹⁵

In accordance with the preferred mode of the present invention, the user can access the underlying MAPPER data manipulation capabilities in a JavaScript object-based programming environment. Therefore, programmers knowledgeable in the practices of standard programming languages such as JavaScript can readily apply those skills to utilize the data manipulation and other capabilities derived from the underlying MAPPER engine. Each JavaScript represents a stored procedure of varying degrees of complexity that can be called from various development and application software within the DACS BISNET product suite. Previously, these MAPPER engine capabilities were available using the proprietary MAPPER run-script procedural language.¹⁶

In the preferred implementation, the JavaScript parser and objects are integrated into the MAPPER engine to support JavaScript stored procedures. The integrated JavaScript parser interprets and executes JavaScript stored procedures, which utilize custom JavaScript objects. These custom capabilities in an object-based, paradigm for dataset manipulation and analysis purposes. Additional custom JavaScript objects are also provided to support the more complex MAPPER core engine "power" function analysis

¹⁵See Specification at page 10, lines 18-22.

¹⁶See Specification at page 11, lines 1-8.

capabilities. JavaScript stored procedures are an alternative to MAPPER run-script, input and output arguments can be passed, and a resulting dataset can be returned to the caller.¹⁷

A key to making this process efficient is the technique for "parameterization" of the underlying MAPPER "power" commands. In order to leverage the more complex MAPPER core engine "power" function analysis capabilities, it is necessary for the programmer to supply a set of arguments. The arguments are positional and the number can range from just a few to many dozens. As the number of arguments increases, the burden of programming them can become unmanageable.¹⁸

As originally conceived, the MAPPER engine power functions were invoked via the procedural MAPPER run-script language. This interface is satisfactory for programming simple sets of arguments, although it has the inherent disadvantage of requiring intricate knowledge of the proprietary MAPPER run-script language syntax. This syntax is very efficient, but at the tradeoff of being cryptic and therefore error prone and requiring specialized training. As the number of arguments increases, the programming task becomes daunting.¹⁹

¹⁷See Specification at page 11, lines 9-16.

¹⁸See Specification at page 11, lines 17-22.

¹⁹See Specification at page 12, lines 1-6.

To compliment the JavaScript Dataset object, which represents a physical MAPPER database table, a suite of Parameter objects is provided to allow programming the numerous combinations of arguments that parameterize the processing performed by MAPPER core engine power function analysis functions. A separate JavaScript Parameter object is provided for each of the MAPPER core engine power functions. Each parameter object contains custom properties, methods , and compound objects that conform to the programming requirements of a specific power function.²⁰

The preferred mode is preferable to prior art approaches because the parameterization of the MAPPER engine power functions is presented in a JavaScript object-based paradigm. This programming paradigm is readily discernable to programmers that are knowledgeable in modern programming languages and disciplines. Furthermore, it does not require programming knowledge in the proprietary MAPPER procedural run-script language. In addition, it allows programming of the underlying MAPPER engine power function data manipulation, aggregation, and analysis capabilities to be written and structured in an object-based paradigm. Therefore, such programs are easier for other programmers to comprehend enhance, and maintain.²¹

²⁰See Specification at page 12, lines 7-13.

²¹See Specification at page 12, lines 14-21.

The efficiency is further increased with particular attention to the formulation of the resultant reports. The preferred system provides the JavaScript application developer with the capability to select the outcome format generation of a Dataset. This means the elimination of the performance expense of adding additional logic steps within an application. As a result, the system does not need additional performance resources when processing a dataset using a power function, that is a MAPPER core engine function that produces another dataset based on analysis of one or more input datasets. For example, the @SRH core engine function can produce a dataset result containing records that match specified criteria in any input dataset. This allows the application developer the ability to specify the outcome format of the Dataset after the execution of a JavaScript power function with a single specification.²²

Most power functions produce results in a dataset (e.g., records are sorted, columns are updated with calculated results, only records that match certain criteria are included, etc.). The object containing the dataset results is controlled by the target dataset "overwrite" property. The dataset "overwrite" property is loosely related to the access mode of the dataset. If the access mode is "ReadOnly", then the "overwrite" property is always false

²²See Specification at page 12, line 22, through page 13, line 9.

and cannot be changed from JavaScript. Otherwise the overwrite property can be changed from JavaScript. When a power function is executed by a target dataset, the results overwrite the contents of the target dataset. If the overwrite property is true. The results are returned in a new temporary dataset, if the overwrite property is false.²³

Fig. 1 is a pictorial diagram of hardware suite 10 of the preferred embodiment of the present invention²⁴. Fig. 2 is a detailed flow diagram showing integration of JavaScript with the MAPPER engine²⁵. Fig. 8 is a detailed flow diagram showing a particular example in accordance with the preferred mode²⁶.

Claims 11 and 13 are the only pending claims introducing "means-plus-function" limitations. Independent claim 11 has five such limitations which are correlated to Applicants' disclosure as follows:

- a. "permitting means for permitting a user to transfer a service request defined by a standardized object-based programming language"²⁷;
- b. "offering means located within said hardware server responsively coupled to said permitting means via said publicly accessible

²³See Specification at page 13, lines 10-18.

²⁴See Specification at page 15, lines 10-11.

²⁵See Specification at page 17, lines 2-3.

²⁶See Specification at page 24, lines 2-3.

²⁷See Specification at page 15, lines 11-15, and Fig. 1, element 12.

digital data communication network for offering legacy data base management services involving access to at least one dataset having a non-standard scripted command language and which cannot directly execute said standardized object-based programming language"²⁸;

c. "converting means responsively located within said offering means for converting said service request from said standardized object-base programming language to said non-standardized scripted command language"²⁹;

d. "modifying means responsively coupled to said offering means for modifying said dataset if so indicated by said service request"³⁰; and

e. "providing means for providing said resultant report to said user"³¹.

Claim 13 is limited by "means located within said permitting means for generating a second service request"³².

Applicants herewith endeavor to map claims 1, 6, and 21 to "the specification by page and line number, paragraph number, and to the drawings, if any".

Claim 1:--- element a -- see Fig. 1, element 12, and specification at page 15, lines 11-15;

²⁸See Specification at page 15, lines 16-17, and Fig. 1, elements 20 and 22.

²⁹See Specification at page 17, lines 3-5, and Fig. 2, elements 38 and 42.

³⁰See Specification at page 17, lines 7-8, and Fig. 2, element 46.

³¹See Specification at page 16, lines 10-11, and Fig. 1, element 14.

³²See Specification at page 15, lines 14-15, and Fig. 1 element 12.

--- element b -- see Fig. 1, elements 20 and 22, and specification at page 15, lines 16-17;

--- element c -- see Fig. 2, elements 38 and 42, and specification at page 17, lines 3-5; and

--- element d -- see Fig. 1, element 14, and Fig. 2, element 46, and specification at page 16, lines 10-11, and page 17, lines 7-8.

Claim 6:

--- element a -- see Fig. 1, element 12, and specification at page 15, lines 11-15;

--- element b -- see Fig. 1, elements 20 and 22, and specification at page 15, lines 16-17;

--- element c -- see Fig. 2, elements 38 and 42, and specification at page 17, lines 3-5;

--- element d -- see Fig. 2, element 46, and specification at page 17, lines 7-8; and

--- element e -- see Fig. 1, element 14, and Fig. 2, element 46, and specification at page 16, lines 10-11, and page 17, lines 7-8.

Claim 16:

--- element a -- see Fig. 1, element 12, and specification at page 15, lines 11-15;

--- element b -- see Fig. 1, elements 20 and 22, and specification at page 15, lines 16-17;

--- element c -- see Fig. 2, elements 38 and 42, and specification at page 17, lines 3-5; and
--- element d -- see Fig. 2, element 46, and specification at page 17, lines 7-8.

Claim 21:

--- element a -- see Fig. 1, element 12, and specification at page 15, lines 11-15;
--- element b -- see Fig. 1, elements 20 and 22, and specification at page 15, lines 16-17;
--- element c -- see Fig. 2, elements 38 and 42, and specification at page 17, lines 3-5; and
--- element d -- see Fig. 1, element 14, and Fig. 2, element 46, and specification at page 16, lines 10-11, and page 17, lines 7-8.

GROUND OF REJECTION TO BE REVIEWED ON APPEAL

1. Is claim 11 objectionable for "informal invocation of "means-plus-function" language?

2. Are claims 1-21 unpatentable under 35 U.S.C. 102(b) as anticipated by U.S. Patent Application Publication No. 2003/0051070, published in the name of Shappir et al. (hereinafter referred to as "Shappir")?

3. Are claims 1-20 unpatentable under 35 U.S.C. 103(a) as obvious over U.S. Patent Application Publication No. 2005/0192851 published in the name of Rangnekar (hereinafter referred to as "Rangnekar") in view of U.S. Patent No. 5,806,067, issued to Connor (hereinafter referred to as "Connor")?

4. Is claim 21 unpatentable under 35 U.S.C. 103(a) as obvious over Rangnekar in view of Connor and further in view of U.S. Patent No. 6,141,759, issued to Braddy (hereinafter referred to as "Braddy")?

ARGUMENT

I. Claim 11 is not objectionable for "informal invocation of "means-plus-function" language.

Claim 11 has been objected to "for informal invocation of "means" plus function language because it does not recite "means for" clause in steps b-d". This objection is respectfully traversed as not understood and as based upon clearly erroneous findings of fact.

In making his rejection, the Examiner states:

This claim is objected to under 37 CFR 1.75 for not providing antecedent basis for the "means" to (sic) the instant specification because the informal language and argument create an ambiguity whether the claim elements a through c invoke 6th paragraph of 35 USC 112.
"

Claim element a reads:

permitting means for permitting a user to transfer a service request defined by a standardized object-based programming language; (emphasis added)

It is not understood why this element does not define a "means-plus-function" element under the authority of 35 U.S.C. 112, sixth paragraph.

Similarly, it is not understood why the "means for offering" of claim element b and the "means for converting" of claim element c do not comport with the basic rules of English grammar and the requirements of 35 U.S.C. 112, sixth paragraph for defining a limiting element. Furthermore, in accordance with the rules governing the filing of Appeal Briefs, these elements are specifically mapped to Applicants' specification and drawings in the above Summary of the Invention. Thus, the objection to claim 11 should be reversed.

II. Claims 1-21 are not unpatentable under 35 U.S.C. 102(b) as anticipated by Shappir.

Claims 1-21 have been rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent Application No. 2003/0051070, published in the name of Shappir et al. (hereinafter referred to as "Shappir"). The ground of rejection is respectfully traversed as to claims 1-21 for the following reasons.

The standards for a finding of anticipation during examination are specified in MPEP 2131, which provides in part:

TO ANTICIPATE A CLAIM, THE REFERENCE MUST TEACH
EVERY ELEMENT OF THE CLAIM

"A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d

1051, 1053 (Fed. Cir. 1987). "The identical invention must be shown in as complete detail as is contained in the ... claim." *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989). (emphasis added)

The rejection is respectfully traversed because "the identical invention" is not shown by Shappir "in as complete detail as is contained in the claims", as is required by MPEP 2131.

Applicants' invention as disclosed and claimed provides a method and apparatus for requesting services from a legacy data base management system utilizing a computer terminal using standardized object-oriented command language which is incompatible with the legacy data base management system. This is accomplished by using the legacy data base management system to perform the required conversions using parameterized inputs defined in the incompatible standardized object-oriented command language. It is the power of the legacy data base management system which is utilized to perform the conversion. The importance and efficiencies of this technique are discussed throughout Applicants' disclosure, as summarized above. Independent claims 1, 6, 11, 16, and 21 are all limited by this key feature.

For whatever reason, the Examiner has cited and sought to apply Shappir, which performs the conversions external to the legacy system, thereby failing to achieve the advantages of Applicants' approach. The Examiner readily admits this distinction in citing Fig. 1 and paragraph [0066] of Shappir showing the need

to add task server 50 and emulation application 30, to provide an interface to unmodified legacy application 10. In other words, unlike Applicants' invention as disclosed and claimed, Shappir requires the additional developmental, recurring, and run-time costs associated with these added hardware and software additions. Applicants' approach permits the existing legacy data base management system to perform the needed conversion.

As a result of this fundamental distinction in approach of Shappir and Applicants, the pending claims are readily distinguishable over the Shappir disclosure. For example, instead of incurring the costs associated with task server 50 and emulation application 30, many of Applicants' pending claims are limited by various parameters submitted with the service request to control conversion of the command language by the legacy data base management system. Shappir, of course, does not utilize such parameters, because it does not perform conversion within the legacy system.

II.A. Claim 1 is not anticipated by Shappir.

Claim 1 has four basic elements. The second claimed element is "a legacy data base management system including a hardware server which cannot execute said standardized object-based command language responsively coupled to said terminal computer which

honors said user request by execution of a non-standardized command language to produce a result from a dataset within said data base". In making his rejection, the Examiner apparently cites at least 12 full paragraphs of Shappir. However, nowhere in all of that material is there any disclosure of the claimed "legacy data base management system" which executes "a non-standardized command language" and which produces "a result from a dataset within said data base".

As a matter of fact, Shappir says nothing of the execution of a command language, either standard or non-standard. Instead, the Examiner makes a parenthetical reference to SQL, which is means Standardized Query Language. Clearly, SQL is neither "non-standardized", as claimed, nor is it "executed", as claimed.

The third claimed element is "a conversion facility for conversion of said standardized object-based command language to said non-standardized command language which is executable by said legacy data base management system". In making his rejection, the Examiner ignores the claimed limitations citing paragraphs 09658-0070, which establish the any "conversion" alleged to be performed by Shappir is performed external to the legacy system.

The fourth claimed element is "a facility responsively coupled to said legacy data base management system which prepares said result for transfer to said terminal computer and which modifies

said dataset if and only if specified in said service request". Ignoring these limitations, the Examiner cites paragraphs of Shappir which have nothing to do with the claimed invention. The reference says nothing of the claimed "result", nothing of the claimed "transfer" and nothing of the claimed conditional modification to "said dataset".

Furthermore, as explained in Applicants' Abstract:

The approach is particularly efficient in that the user can easily specify the extent to which the dataset associated with honoring of the service request will be left in its original state or a modified state.

Thus, the conditional modification of the accessed "dataset", is not a mere matter of design choice, but it provides in a single service request/response that which would require at least multiple service requests in the system of Shappir. Apparently, the Examiner simply ignores this feature.

As a result of Rangnekar having none of at least three of the four elements of claim 1, the rejection of claim 1, and all claims depending therefrom, should be reversed.

II.B. Claim 2 is not anticipated by Shappir.

Claim 2 depends from claim 1 and further limits the coupling of the claimed "computer terminal" to the claimed "legacy data base management system". Because Shappir does not have the claimed "legacy data base management system" as explained above, it cannot

have the claimed coupling. The rejection of claim 2 should be reversed.

II.C. Claim 3 is not anticipated by Shappir.

Claim 3 depends from claim 2 and is further limited "wherein the user request specifies the dataset". To make his rejection, the Examiner arbitrarily cites paragraphs [0026] and [0070] neither of which saying anything about the claimed limitation.

Paragraph [0026] states in its entirety:

According to another embodiment of the present invention, the format is a format of stored procedures.

Similarly irrelevant s paragraph [0070] which states in full:

In **FIG. 3b**, incompatible legacy systems **300** and **400** are directly interfaced using a server **500** incorporating an API-based emulation task server (server **200** in **FIG. 2**). Using terminal emulation connections **301** and **401**, respectively, systems **300** and **400** automatically exchange data in real time, the operations managed by server **500**.

There is no "dataset", no "user request", no "specifies", etc. in either paragraph. Furthermore, these claimed elements are found nowhere in Shappir. Thus, the Examiner has impermissibly based his rejection on clearly erroneous findings of fact. The rejection of claim 3 should be reversed.

II.D. Claim 4 is not anticipated by Shappir.

Claim 4 depends from claim 3 and further limits the claimed coupling network. As explained above, Shappir does not meet the limitations of claim 3 from which claim 4 depends. Therefore, Shappir cannot meet the further limitations of claim 4. The rejection of claim 4 should be reversed.

II.E. Claim 5 is not anticipated by Shappir.

Claim 5 depends from claim 4 and further limits the claimed "standardized object-based command language". As explained above, Shappir does not meet the limitations of claim 4 from which claim 5 depends. Therefore, Shappir cannot meet the further limitations of claim 5. The rejection of claim 5 should be reversed.

II.F. Claim 6 is not anticipated by Shappir.

Claim 6 is an independent method claim having five limiting steps. Shappir does not have at least four of these five steps. The second claimed element is "receiving said service request by said legacy data base management system". In making his rejection, the Examiner cites paragraph [0021] of Shappir, which says noting of the limitations claimed by Applicants. As explained above, Shappir cannot convert a service request within the legacy system. Therefore, the conversion is made before contact is made with the

legacy system. Thus, there is no "receiving said service request by said legacy data base management system".

The third claimed element is "converting said service request from said standardized object-based command language into said non-standardized command language by said legacy data base management system". In making his rejection, the Examiner again cites paragraph [0066] of Shappir to establish that the claimed converting cannot be performed by the "legacy application" 10, but is rather performed by the externally added "task server" 50 and externally added "emulation application" 30. Shappir simply does not perform conversion within legacy application 10.

The fourth claimed element is "honoring said service request by executing said non-standardized command language to access a dataset from said data base by said legacy digital data base management system". In clearly erroneously making his rejection, the Examiner cites paragraphs [0037] and [0038] of Shappir which does not even mention "executing said non-standardized command language", does not even mention "access a dataset", etc.

The fifth claim element requires conditionally "modifying" a dataset. The Examiner again cites paragraphs [0037] and [0038], which have nothing to do with this limitation. The rejection of claim 6, and all claims depending therefrom, should be reversed.

II.G. Claim 7 is not anticipated by Shappir.

Claim 7 depends from claim 6 and is further limited by "wherein said dataset is specified by said service request". Nowhere in the cited material or elsewhere in Shappir is a "dataset" even mentioned parenthetically. The rejection of claim 7 should be reversed for failure to address the claimed invention.

II.H. Claim 8 is not anticipated by Shappir.

Claim 8 depends from claim 7 and further limits the claimed coupling network. As explained above, Shappir does not meet the limitations of claim 7 from which claim 8 depends. Therefore, Shappir cannot meet the further limitations of claim 8. The rejection of claim 8 should be reversed.

II.I. Claim 9 is not anticipated by Shappir.

Claim 9 depends from claim 8 and further limits the claimed coupling network. As explained above, Shappir does not meet the limitations of claim 8 from which claim 9 depends. Therefore, Shappir cannot meet the further limitations of claim 9. The rejection of claim 9 should be reversed.

II.J. Claim 10 is not anticipated by Shappir.

Claim 10 depends from claim 9 and further limits the claimed "standardized object-based command language". As explained above, Shappir does not meet the limitations of claim 9 from which claim 10 depends. Therefore, Shappir cannot meet the further limitations of claim 10. The rejection of claim 10 should be reversed.

II.K. Claim 11 is not anticipated by Shappir.

Claim 11 is an independent apparatus claim having five "means-plus-function" limitations. Nevertheless as explained above, the Examiner ignores his obligations under MPEP 2181-2184. For example, MPEP 2181 requires the Examiner to explicitly acknowledge these "means-plus-function" limitations. Even though the Examiner has objected to the first three claim elements (i.e., a-c) alleging some sort of informality as discussed above, he has not objected to the remaining "means-plus-function" claim elements. Yet, he has not provided the recognition of MPEP 2181.

Furthermore, he simply states that Shappir meets the claimed elements using copious citations to support his rejection. Yet, these citations do not support the findings for which they are cited.

For example, the first claimed element is a "permitting means for permitting a user to transfer a service request....". As stated above in the Summary of the Invention, this claimed element

reads directly on element 12 of Fig. 1, which is described in the specification at page 15, lines 11-15. Fig. 1, element 12, is clearly identified as a personal computer functioning as an "Internet Terminal". Yet in making his rejection, the Examiner cites paragraphs [0066] and [0070] of Shappir which say nothing of a personal computer functioning as an Internet Terminal, or a reasonable equivalent thereof. Though the citations do reference a software program entitled "terminal emulator" 20, surely no one would consider a computer program to be a reasonable equivalent to a piece of hardware. Furthermore, neither the Examiner nor Shappir indicate how the claimed "user" would utilize the alleged "permitting means" of Shappir.

Similar issues arise which each of the Examiner's citations as support for his finding of Applicants' claimed elements within Shappir. Certainly, he does not assert that the claimed "converting means" is found within legacy application 10. Surely, one could not reasonably argue that paragraph [0070] of Shappir teaches the conditional "modifying" which is limiting of the claimed "modifying means". Nowhere in paragraph [0066] is there any showing of "providing said resultant report to said user" as is claimed.

Thus, the rejection of claim 11 should be reversed.

II.L. Claim 12 is not anticipated by Shappir.

Claim 12 depends from claim 11 and is further limited by the claimed service request specifying the claimed dataset. In making his rejection, the Examiner cites Fig. 1 of Shappir, along with paragraphs [0023] and [0065]. Not only do these citations not support the Examiner's findings, there is no mention of a "dataset". The rejection of claim 12 should be reversed.

II.M. Claim 13 is not anticipated by Shappir.

Claim 13 depends from claim 12 and is further limited by the claimed "means located within said permitting means for generating a second service request". Having not found the claimed "permitting means" in his rejection of claim 11, the Examiner somehow finds that the claimed limitations can somehow be found in Fig. 4b and paragraph [0072] of Shappir. This finding is unsupported by the prior art and is therefore clearly erroneous. The rejection of claim 13 should be reversed.

II.N. Claim 14 is not anticipated by Shappir.

Claim 14 depends from claim 13 and further limits the claimed "offering means". Because Shappir does not have the claimed "offering means" as explained above, it cannot have these further limitations. Therefore, the Examiner simply makes the unsupported

statement that Shappir somehow contains these limitations. The rejection of claim 14 should be reversed.

II.O. Claim 15 is not anticipated by Shappir.

Claim 15 depends from claim 14 and further limits the claimed "permitting means". Because Shappir does not have the claimed "permitting means" as explained above, it cannot have these further limitations. The rejection of claim 15 should be reversed, because Shappir does not meet the standards of MPEP 2131 for a finding of anticipation.

II.P. Claim 16 is not anticipated by Shappir.

Claim 16 has had the limitations of the rather extensive preamble moved into the body of the claim. These limitations provide the basic environment of the invention, along with two key limiting elements. Shappir does not have these environmental limitations for the reasons discussed above. Thus, the Examiner simply finds these limitations without support from the prior art of record.

The third claimed element is the "conversion facility". Because Shappir does not have this element, the Examiner cites unrelated Figs. 1-2 and paragraphs [0066] and [0070]. Though the Examiner is aware that the claimed "conversion facility" is "located within said legacy data base management system", he simply ignores this limitation.

The fourth claimed element is the "facility which modifies" the claimed "dataset". In making his rejection, the Examiner again cites paragraph [0070] which is quoted above. This paragraph says nothing of the claimed conditional modification and does not even parenthetically mention the term "dataset".

It is unknown why the Examiner considers this relevant to the claimed element. Perhaps the Examiner equates the claimed "modifies" with the disclosed "exchange date". If so, this is also logically inconsistent.

Therefore, the rejection of claim 16, and all claims depending therefrom, should be reversed.

II.Q. Claim 17 is not anticipated by Shappir.

Claim 17 depends from claim 16 and is further limited by "wherein said dataset is specified by said service request". Apparently, the Examiner has found this within Shappir completely without support from the reference, because nowhere in the cited material is a "dataset" even mentioned. In fact, Applicants have been unable to find the mention of a "dataset" anywhere in the reference. The rejection of claim 17 should be reversed.

II.R. Claim 18 is not anticipated by Shappir.

Claim 18 depends from claim 17 and further limits the claimed coupling network. Because Shappir cannot meet the limitations of

claim 17 from which claim 18 depends, it cannot meet the further limitations of claim 18. The rejection of claim 18 should be reversed.

II.S. Claim 19 is not anticipated by Shappir.

Claim 19 depends from claim 18 and further limits the claimed coupling network. Because Shappir cannot meet the limitations of claim 18 from which claim 19 depends, it cannot meet the further limitations of claim 19. The rejection of claim 19 should be reversed.

II.T. Claim 20 is not anticipated by Shappir.

Claim 20 depends from claim 19 and further limits the claimed standardized command language. Because Shappir cannot meet the limitations of claim 19 from which claim 20 depends, it cannot meet the further limitations of claim 20. The rejection of claim 20 should be reversed.

II.U. Claim 21 is not anticipated by Shappir.

In rejecting claim 21, the Examiner makes many of the same errors discussed above. In addition, the Examiner makes additional errors. For example, in finding the claimed "computer terminal which generates a user request in a standardized object-based command language which specifies access to a dataset within a data base", the Examiner cites Figs. 1, 2, and 4b, along with the

Abstract and paragraph [0020] of Shappir. Yet, nowhere in Shappir is the claimed "dataset" even mentioned.

Similarly, the second claimed element is a "hardware server containing a legacy data base management system...." and the Examiner cites Fig. 2, element 100, and eight fully paragraphs of text, none of which even referring to Fig. 2, element 100.

The third claimed element is a "conversion facility located within said legacy data base management system....". For whatever reason, the Examiner makes citations to two paragraphs having nothing to do with Fig. 2, element 100, which the Examiner has found to be the "legacy data base management system".

The fourth claimed element is a "facility" which provides a conditional modification to the claimed "dataset". As explained above, Shappir does not have a conditional modification to anything. Furthermore, Shappir never even mentions a "dataset". Therefore, the Examiner cites paragraph [0070], which has no relationship to the claimed invention.. The rejection of claim 21 should be reversed.

III. Claims 1-20 are not unpatentable over Rangnekar in view of Connor.

Claims 1-20 have been rejected under 35 U.S.C. 103(a) as being unpatentable over Rangnekar in view of Connor. The ground of rejection should be reversed for the following reasons.

To make a *prima facie* case of obviousness, MPEP 2143 requires the Examiner to provide evidence and argument showing: 1) motivation to make the alleged combination; 2) reasonable likelihood of success of the alleged combination; and 3) all claimed elements within the alleged combination. The Examiner has failed to make any of these three required showings. Therefore, because the Examiner has not made a *prima facie* case of obviousness, Applicants need not and indeed cannot offer appropriate evidence and argument in rebuttal.

The first required showing is that of motivation. In an apparent attempt to show motivation to combine Connor with Rangnekar, the Examiner states:

A skilled artisan would have been motivated to combine Rangnekar and Connor because it provides for a process for organizing and managing the transition, a multi-tier (sic) client/server architecture that adheres to open systems standards" as discussed in Connor, Abstract.

Quite apart from the fact that this finding is largely incomprehensible, it disingenuously suggests that it somehow relates to, and even quotes from, the Abstract of Connor. This is incorrect. The Abstract of Connor (and indeed the body of the disclosure thereof) is concerned with reconciling ambiguous data fields, such as representing the year as 2008 or 08.

Thus, not only would one not be motivated to make the alleged combination, Rangnekar and Connor have nothing to do with each other. It is inconceivable that the ATM based booking query system of Rangnekar would ever be confronted with such an ambiguity as is

the focus of Connor. AS a result, there is no motivation to make the alleged combination and none has been shown.

The second showing required by MPEP 2143 is that of reasonable likelihood of success. The Examiner simply ignores this requirement, because of the readily apparent incompatibilities between Rangnekar and Connor.

The third showing required by MPEP 2143 is that all claimed elements must be found within the alleged combination. This requirement cannot be satisfied as shown with reference to the individual claims due to the substantial differences of structure and function between Applicants' invention as disclosed and claimed and the alleged combination.

As explained above, Applicants' invention provides a method and apparatus for requesting services from a legacy data base management system utilizing a computer terminal using standardized object-oriented command language which is incompatible with the legacy data base management system. This is accomplished by using the legacy data base management system to perform the required conversions using parameterized inputs defined in the incompatible standardized object-oriented command language.

For whatever reason, the Examiner has cited and sought to apply Rangnekar, which permits a customer to enter a booking query at an ATM. Thus, instead of a user performing data base functions using a data base management system via a user terminal as claimed,

the user of the Rangnekar is limited to the minimal pre-defined functions which can be performed by an ATM. To the extent that the Examiner cites conversions within Rangnekar, these amount to mere conversions of data string message formats within a web server rather than the claimed conversion of "user requests" within the same legacy data base management system as is used to actually honor the claimed service request. As a result of the lack of pertinence of this reference, the rejection of claims 1-20 can only be based upon clearly erroneous findings of fact and incorrect application of controlling law.

A key difference is that Rangnekar submits data via an ATM machine which may be converted, whereas Applicants' claimed invention submits command language which must be converted by the legacy data base management system into command language script which is executable by the legacy data base management system. Rangnekar converts data whereas Applicants' invention converts executable command language script. As is limiting of the pending dependent claims, various parameters submitted with the service request control conversion of the command language by the legacy data base management system.

Connor appears to be a mere software invention directed to the use of source code analysis to resolve ambiguities within data bases which use different conventions for representing identity of a year. Thus, Connor is even less pertinent to Applicants'

invention Rangnekar. This lack of pertinence becomes even more prominent as one considers the rejections of the individual claims.

III.A. Claim 1 is not unpatentable over Rangnekar in view of Connor.

Claim 1 has four basic elements. The first element is "a terminal computer operable by a user which generates a user request in a standardized object-based command language for access to a data base". In making his rejection, the Examiner cites an "automatic teller machine" of Rangnekar. As a matter of law, the Examiner cannot find that the claimed "terminal computer" is "interpreted to include "ATM....End User" as alleged. There is no disclosure that the ATM of Rangnekar is a "computer" as claimed.

By way of definition, page 15, lines 11-13, Applicants' specification states:

The client interfaces with the system via Internet terminal 12. Preferably Internet terminal 12 is an industry compatible, personalized computer having a current version of the Windows operating system and suitable web browser, all being readily available commercial products.

It is inconceivable that this claimed element could be "interpreted to include ATM" as alleged. Because the ATM is not the claimed terminal, it cannot generate "a user request in a standardized object-based command language for access to a data base" as

claimed. Actually, the user of an ATM can merely supply data (e.g., account number, PIN, etc.).

The second claimed element is "a legacy data base management system including a hardware server which cannot execute said standardized object-based command language responsively coupled to said terminal computer which honors said user request by execution of a non-standardized command language to produce a result from a dataset within said data base". In making his rejection, the Examiner apparently cites CRS 30 of Rangnekar. However, the Examiner ignores the remainder of the claim which requires the claimed "legacy data base management system" to be "responsively coupled to said computer terminal". Surely, the Examiner does not suggest that Rangnekar discloses that CRS 30 is somehow "responsively coupled to said computer terminal", as claimed. Furthermore, because Rangnekar says nothing of a "dataset" within the claim associated with CRS 30, Examiner simply ignores the claimed limitation. In addition, the claim requires that the "legacy data base management system" honors the claimed user request by execution of a non-standardized command language to produce a result from a dataset. Nowhere does Rangnekar suggest that CRS 30 meets these limitations. Therefore, the Examiner simply ignores them.

Notwithstanding his own findings, the Examiner states:

However Rangnekar does not expressly teach "including a hardware server which cannot execute said standardized object-based command language responsive (sic) coupled to said terminal computer", standardized command language which is executable by said legacy database management system.

Clearly, this finding is incomprehensible. However, it appears that the Examiner has attempted to apply Connor as follows:

Connor teaches "including a hardware server which cannot execute said standardized object-based command language responsive (sic) coupled to said terminal computer". . . . (emphasis added)

Whereas there is insufficient disclosure within Connor to support or deny that it "cannot execute said standardized object-based command language", it is clear that Connor discloses a single processor system (see Fig. 1). Therefore, the allegation that Connor is "coupled to said terminal computer" is clearly erroneous.

The third claimed element is "a conversion facility for conversion of said standardized object-based command language to said non-standardized command language which is executable by said legacy data base management system". In making his rejection, the Examiner ignores the claimed limitations and instead citing Figs. 18-19 and paragraphs 0118-0119. As if any of these citations have anything to do with the claimed element or HTML, the Examiner impermissibly states including "HTML". As explained above, the claim element clearly requires conversion of the command language of the service request, rather than the data of the Rangnekar data

transfer. Applicants' invention as disclosed and claimed converts the service request whereas Rangnekar merely converts data.

The fourth claimed element is "a facility responsively coupled to said legacy data base management system which prepares said result for transfer to said terminal computer and which modifies said dataset if and only if specified in said service request".

Ignoring these limitations, the Examiner states:

....including "XML document is updated at a financial services system server only if there is a change in the city data"....

The Examiner then cites five full paragraphs of Rangnekar which have nothing to do with the claimed invention. It is truly confusing why the Examiner would consider these citations related to claim 1. Applicants' invention converts the claimed "service request", not "data".

As a result of Rangnekar having none of the four elements of amended claim 1, the rejection of claim 1, and all claims depending therefrom, should be reversed.

III.B. Claim 2 is not unpatentable over Rangnekar in view of Connor.

Claim 2 depends from claim 1 and further limits the coupling of the claimed "computer terminal". Even though the Examiner found the "computer terminal" of claim 1 to be "interpreted to be an ATM", he finds the further limitations to the coupling to be in the

totally unrelated paragraphs 7 and 11 of Rangnekar. The claim explicitly requires the claimed "computer terminal" to be coupled in the claimed fashion. Yet, the Examiner ignores Rangnekar, paragraph 0111, which begins:

As illustrated in FIG. 14, financial institutions run their ATM's 12 on their private networks.

Thus the Examiner's findings are incorrect as a matter of law.

Nevertheless, the Examiner has previously responded stating:

The argument that the network is private does not necessarily contradict the network being "publicly accessible".

This statement is legally irrelevant, because MPEP 2143 is concerned with that which is disclosed by the reference. The law requires disclosure of the claimed element. The argument made by the Examiner is neither disclosed by Rangnekar or Connor, nor supported by the disclosure of Rangnekar and Connor. The rejection of claim 2 should be reversed.

III.C. Claim 3 is not unpatentable over Rangnekar in view of Connor.

Claim 3 depends from claim 2 and is further limited "wherein the user request specifies the dataset". To make his rejection, the Examiner arbitrarily cites paragraphs 8-9 of Rangnekar, which say nothing of the claimed limitation. There is no "dataset", no ATM, etc. Thus, the Examiner has impermissibly based his rejection

on clearly erroneous findings of fact. The rejection of claim 3 should be reversed.

III.D. Claim 4 is not unpatentable over Rangnekar in view of Connor.

Claim 4 depends from claim 3 and further limits the claimed coupling network. As explained above, the alleged combination does not meet the limitations of claim 3 from which claim 4 depends. Therefore, the alleged combination cannot meet the further limitations of claim 4. Thus, the Examiner ignores the claimed invention and his previous findings to cite Rangnekar, paragraph 11 which is throughly unrelated to his previous finding. Claim 4 requires coupling of specific components in a particular manner. The rejection of claim 4 should be reversed.

III.E. Claim 5 is not unpatentable over Rangnekar in view of Connor.

Claim 5 depends from claim 4 and further limits the claimed "standardized object-based command language". To make his rejection, the Examiner again ignores the claimed invention and simply seeks to find the words of the claim in disparate and unrelated portions of the prior art. He cites paragraph 107, which relates to software not found in CRS 30, or the ATM 12, or

processor 105 of Connor. The rejection of claim 5 should be reversed.

III.F. Claim 6 is not unpatentable over Rangnekar in view of Connor.

Claim 6 is an independent method claim having five limiting steps. The alleged combination of Rangnekar and Connor has none of these five steps. Apparently, the Examiner has not found the first step to be expressly disclosed by the alleged combination, so the Examiner states:

....including "prints your itinerary....

This statement is clearly erroneous on its face. Furthermore, the statement does not address Applicants' claimed invention, so even if true, it is legally irrelevant. In fact, it is unknown why some one would consider "prints your itinerary" to be pertinent to the first claimed element. Therefore, the first element of claim 6 is admittedly not found in the alleged combination.

The second claimed element is "receiving said service request by said legacy data base management system". In making his rejection, the Examiner cites paragraph 0092 of Rangnekar, which concerns operation of a web server neither claimed by Applicants nor associated with the basic system having a "computer terminal" and a "legacy data base management system".

The third claimed element is "converting said service request from said standardized object-based command language into said non-standardized command language by said legacy data base management system". In making his rejection, the Examiner states:

....where CRS is a legacy system such as shown above)....

In clearly erroneously finding the fourth element, the Examiner again admits that the alleged combination has no "legacy data base management system" which receives the claimed "service request" for conversion as required by claim element b. Furthermore, he readily admits that the conversion is not made by CRS 30 of Rangnekar.

The fifth claim element requires conditionally "modifying" a dataset. The Examiner cites paragraph 0150 of Rangnekar which has nothing to do with this limitation. The rejection of claim 6, and all claims depending therefrom, should be reversed.

III.G. Claim 7 is not unpatentable over Rangnekar in view of Connor.

Claim 7 depends from claim 6 and further limited by "wherein said dataset is specified by said service request". Instead of addressing the claimed limitation, the Examiner appears intent on showing the disclosure of a highlighted portion of Fig. 25 of Rangnekar. The rejection of claim 7 should be reversed for failure to address the claimed invention.

III.H. Claim 8 is not unpatentable over Rangnekar in view of Connor.

Claim 8 depends from claim 7 and further limits the claimed coupling network. As explained above, the alleged combination does not meet the limitations of claim 7 from which claim 8 depends. Therefore, the alleged combination cannot meet the further limitations of claim 8. And again, the Examiner ignores the first sentence of Rangnekar, paragraph 0111, which shows that even if ATM 12 did generate the claimed "service request", it is expressly coupled to a "private network". The rejection of claim 8 should be reversed.

III.I. Claim 9 is not unpatentable over Rangnekar in view of Connor.

Claim 9 depends from claim 8 and further limits the claimed coupling network. As explained above, Connor discloses a single processor system and Rangnekar explicitly discloses coupling only via a "private network" (see paragraph 0111). The rejection of claim 9 should be reversed.

III.J. Claim 10 is not unpatentable over Rangnekar in view of Connor.

Claim 10 depends from claim 9 and further limits the claimed "standardized object-based command language". To make his

rejection, the Examiner has again made a finding completely unsupported by the prior art of record. The rejection of claim 10 should be reversed.

III.K. Claim 11 is not unpatentable over Rangnekar in view of Connor.

Claim 11 is an independent apparatus claim having five "means-plus-function" limitations. Nevertheless as explained above, the Examiner ignores his obligations under MPEP 2181-2184. For example, MPEP 2181 requires the Examiner to explicitly acknowledge these "means-plus-function" limitations. Furthermore, he has clearly refused to apply prior art teachings which are the same as or reasonable equivalents of the components disclosed by Applicants as required by law. Thus, the rejection of claim 11 should be reversed for absence of examination in accordance with controlling law in addition to the clearly erroneous findings of fact and clear errors of law discussed above.

III.L. Claim 12 is not unpatentable over Rangnekar in view of Connor.

Claim 12 depends from claim 11 and is further limited by the claimed service request specifying the claimed dataset. In making his rejection, the Examiner cites Fig. 22 of Rangnekar, showing "options for travel between Bangalore-Delhi" (see paragraph 0070),

which has nothing to do with the claimed "service request" or the claimed "dataset". The rejection of claim 12 should be reversed.

III.M. Claim 13 is not unpatentable over Rangnekar in view of Connor.

Claim 13 depends from claim 12 and is further limited by the claimed "means located within said permitting means for generating a second service request". Having not found the claimed "permitting means" in his rejection of claim 11, the Examiner somehow finds that the claimed limitations can somehow been found in Figs. 22-23 of Rangnekar. This finding is unsupported by the prior art and is therefore clearly erroneous. The rejection of claim 13 should be reversed.

III.N. Claim 14 is not unpatentable over Rangnekar in view of Connor.

Claim 14 depends from claim 13 and further limits the claimed "offering means". Because the alleged combination does not have the claimed "offering means", it cannot have these further limitations. Therefore, the Examiner simply makes the statement that Rangnekar somehow contains these limitations even though unsupported by the cited paragraph 0207. The rejection of claim 14 should be reversed.

III.O. Claim 15 is not unpatentable over Rangnekar in view of Connor.

Claim 15 depends from claim 14 and further limits the claimed "permitting means". Because the alleged combination does not have the claimed "permitting means" as explained above, it cannot have these further limitations. Therefore, the Examiner irrelevantly cites Rangnekar, paragraph 0170. This material is legally irrelevant, because the claim is limiting of the "permitting means", which the Examiner has found to be an ATM of Rangnekar. Yet, paragraph 0170 has nothing to do with the ATM. Thus, the citation is legally irrelevant. The rejection of claim 15 should be reversed.

III.P. Claim 16 is not unpatentable over Rangnekar in view of Connor.

Claim 16 is an independent apparatus claim having four limiting elements. The issues associated with the first two claimed elements are discussed above.

The third claimed element is the "conversion facility". Because the alleged combination does not have this element, the Examiner cites unrelated paragraph 0207. Apparently, for some reason, the Examiner appears to assume that "Perl using COM" is somehow related to the claimed "conversion facility". In addition

to these findings being unrelated to the claimed invention, this finding is clearly erroneous.

The fourth claimed element is the "facility which modifies" the claimed "dataset" conditioned on an indication within the claimed service request. In making his rejection, the Examiner cites paragraph 0220 stating:

"Cancelled (sic) means that this transaction was cancelled (sic) upon the customer's request".

It is unknown why the Examiner considers this relevant to the claimed element. Perhaps the Examiner equates the claimed "modifies" with the disclosed "canceled". If so, this is also logically inconsistent.

Therefore, the rejection of claim 16, and all claims depending therefrom, should be reversed.

III.Q. Claim 17 is not unpatentable over Rangnekar in view of Connor.

Claim 17 depends from claim 16 and is further limited by "wherein said dataset is specified by said service request". Apparently, the Examiner has found this within Rangnekar completely without support from the reference. The rejection of claim 17 should be reversed.

III.R. Claim 18 is not unpatentable over Rangnekar in view of Connor.

Claim 18 depends from claim 17 and further limits the claimed coupling network. Because the alleged combination cannot meet the limitations of claim 17 from which claim 18 depends, it cannot meet the further limitations of claim 18. Therefore, the Examiner cites a "disembodied" reference to "Internet" 12 at Fig. 2 of Rangnekar, which is clearly not coupled to ATM 12 (see paragraph 0111 in addition to Fig. 2). That the reference parenthetically mentions "Internet" which is not coupled as required by the claim, is legally irrelevant. The rejection of claim 18 should be reversed.

III.S. Claim 19 is not unpatentable over Rangnekar in view of Connor.

Claim 19 depends from claim 18 and further limits the claimed coupling network. Because the alleged combination cannot meet the limitations of claim 18 from which claim 19 depends, it cannot meet the further limitations of claim 19. The rejection of claim 19 should be reversed.

III.T. Claim 20 is not unpatentable over Rangnekar in view of Connor.

Claim 20 depends from claim 19 and further limits the claimed standardized command language. Because the alleged combination

cannot meet the limitations of claim 19 from which claim 20 depends, it cannot meet the further limitations of claim 20. The rejection of claim 20 should be reversed.

IV. Claim 21 is not unpatentable over Rangnekar in view of Connor and further in view of Braddy.

In rejecting claim 21, to the unmotivated and incompatible alleged combination of Rangnekar and Connor, the Examiner alleges the further combination with Braddy. In an apparent attempt to allege motivation for this further combination, the Examiner states:

A skilled artisan would have been motivated to combine Rangnekar in view of Connor and Braddy because it provides for file type extension mappings are used to map file extension to execution path as suggested in Braddy, Fig 6a, item 86.

This finding is truly incomprehensible. However, it does seem clear that Braddy discloses nothing which would complement the ATM system of Rangnekar or the single computer of Connor.

Again, the Examiner ignores his obligation to show reasonable likelihood of success, because of the readily apparent incompatibilities. The Examiner similarly fails to show all of the claimed elements within the alleged combination as required by MPEP 2143.

In attempting to do so, the Examiner makes many of the same errors discussed above. In addition, the Examiner makes additional

errors. For example, in finding the claimed "computer terminal which generates a user request in a standardized object-based command language which specifies access to a dataset within a data base", the Examiner makes much of the ".pl" in the Window bar of Figs. 16-17. Because the reference does not disclose the implications of ".pl", the Examiner now provides Braddy concerning what Rangnekar could have disclosed. Basically, the Examiner asserts that the ".pl" file extension is "used to map file extension to execution path". It is unknown why the Examiner finds that this somehow relates to the claimed "terminal which generates a user request....".

Similarly, the second claimed element is a "legacy data base management system...." and the Examiner cites Fig. 12 showing an ATM. The Examiner further cryptically mentions Phone Agents 2-5 and "travel desk" of Fig. 5B.

The third claimed element is a "conversion facility". For whatever reason, the Examiner makes citations concerning PERL, but refuses to address "located within said legacy data base management system". Apparently, this limitation is ignored, because it is admittedly not found in the prior art of record.

The fourth claimed element is a "facility" providing for conditional modification of the dataset. In making his rejection, the Examiner cites four unrelated paragraphs from Rangnekar. The rejection of claim 21 should be reversed.

CONCLUSION

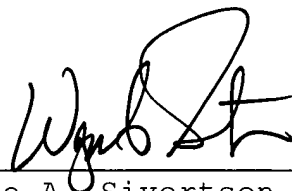
Having thus reviewed the final rejections of claims 1-21, being all pending claims, it seems abundantly clear that the limitations of these claims are not unpatentable in view of the prior art of record. Thus, the rejection of these claims should be reversed as being based upon clearly erroneous fact findings and errors of law.

Respectfully submitted

Barbara A. Christensen et al.

By their attorney,

Date December 15, 2008



Wayne A. Sivertson
Reg. No. 25,645
Suite 401, Broadway Place East
3433 Broadway Street N.E.
Minneapolis, Minnesota 55413
(612) 331-1464

CLAIMS APPENDIX

1. An apparatus comprising:
 - a. a terminal computer operable by a user which generates a user request in a standardized object-based command language for access to a data base;
 - b. a legacy data base management system including a hardware server which cannot execute said standardized object-based command language responsively coupled to said terminal computer which honors said user request by execution of a non-standardized command language to produce a result from a dataset within said data base;
 - c. a conversion facility for conversion of said standardized object-based command language to said non-standardized command language which is executable by said legacy data base management system; and
 - d. a facility responsively coupled to said legacy data base management system which prepares said result for transfer to said terminal computer and which modifies said dataset if and only if specified in said service request.

2. The apparatus of claim 1 wherein said terminal computer is coupled to said legacy data base management system via a publicly accessible digital data communication network.

3. The apparatus of claim 2 wherein said user request specifies said dataset.

4. The apparatus of claim 3 wherein said publicly accessible digital data communication network further comprises the Internet.

5. The apparatus of claim 4 wherein said standardized object-based command language further comprises a commonly available command language.

6. A method of utilizing a terminal using a standardized object-based command language to access a legacy data base management system having a data base employing a non-standardized command language and which cannot execute said standardized object-based command language comprising:

- a. transmitting a service request in said standardized object-based command language from said terminal requesting access to said data base of said legacy data base management system;

- b. receiving said service request by said legacy data base management system;
- c. converting said service request from said standardized object-based command language into said non-standardized command language by said legacy data base management system;
- d. honoring said service request by executing said non-standardized command language to access a dataset from said data base by said legacy digital data base management system; and
- e. modifying said dataset if indicated by said service request.

7. A method according to claim 6 wherein said dataset is specified by said service request.

8. A method according to claim 7 wherein said transmitting step occurs over a publicly accessible digital data communication network.

9. A method according to claim 8 wherein said publicly accessible digital data communication network further comprises the Internet.

10. A method according to claim 9 wherein said standardized object-based command language further comprises a commonly used command language.

11. An apparatus for providing access to a hardware server hosting a legacy data base management systems from a computer terminal using a standardized object-based programming language to efficiently provide a resultant report comprising:

- a. permitting means for permitting a user to transfer a service request defined by a standardized object-based programming language;

- b. offering means located within said hardware server responsively coupled to said permitting means via said publicly accessible digital data communication network for offering legacy data base management services involving access to at least one dataset having a non-standard scripted command language and which cannot directly execute said standardized object-based programming language;

- c. converting means responsively located within said offering means for converting said service request from said standardized object-base programming language to said non-standardized scripted command language;

d. modifying means responsively coupled to said offering means for modifying said dataset if so indicated by said service request; and

e. providing means for providing said resultant report to said user.

12. An apparatus according to claim 11 wherein said dataset is specified by said service request.

13. An apparatus according to claim 12 further comprising means located within said permitting means for generating a second service request.

14. An apparatus according to claim 13 wherein said offering means further comprises a commercially available data base management system.

15. An apparatus according to claim 14 wherein said permitting means further comprises an industry standard personal computer.

16. A data processing system comprising:

a. a terminal computer which generates a service request in a standardized object-based command language;

- b. a hardware server hosting a legacy data base management system which accesses a dataset to honor said service request by executing a non-standardized command language responsively coupled to said terminal and which cannot execute said standardized object-based command language;
- c. a conversion facility located within said legacy data base management system which converts said service request from said standardized object-based command language to said non-standardized command language; and
- d. a facility which modifies said dataset only if indicated by said service request.

17. The data processing system according to claim 16 wherein said dataset is specified by said service request.

18. The data processing system according to claim 17 wherein said terminal computer is responsively coupled to said legacy data base management system via a publicly accessible digital data communication network.

19. The data processing system according to claim 18 wherein said publicly accessible digital data communication network further comprises the Internet.

20. The data processing system according to claim 19 wherein said standardized object-based command language further comprises a commonly utilized command language.

21. An apparatus for accessing a database comprising:

a. a computer terminal which generates a user request in a standardized object-based command language which specifies access to a dataset within a data base;

b. a hardware server containing a legacy data base management system which cannot execute said standardized object-based command language responsively coupled to said terminal computer via a publicly accessible digital data communication network which honors said user request by execution of a non-standardized command language to produce a result from said dataset;

c. a conversion facility located within said legacy data base management system for conversion of said standardized object-based command language to said non-standardized command language which is executable by said legacy data base management system; and

d. a facility responsively coupled to said legacy data base management system which prepares said result for transfer to said terminal computer and which modifies said dataset if and only if specified in said service request.

EVIDENCE APPENDIX

There is no evidence or documents deemed appropriate to be included within this Appendix.

RELATED PROCEEDINGS APPENDIX

There are no decisions or other papers deemed appropriate to be included in this Appendix.